Upgrade Package for LCS with LAHET2.7

Richard E. Prael
Group XCI
Los Alamos National Laboratory

LA-UR-97-4981
December 1, 1997

Introduction

In conjunction with the release of MCNP Version 4B [2], a new scheme for integrating MCNP into the LAHET Code System [1] has been developed. In this method, low-energy neutrons from LAHET and photons from PHT are passed to MCNP in the format of a "surface source read (SSR)" RSSA file. A code TRANSM was created to replace MRGNTP, combining the file merge feature with translation from NEUTP file format to RSSA file format.

Surface crossing data from MCNP may be passed back to HTAPE for tallying in the format of an MCNP "surface source write (SSW)" WSSA file. In addition, HTAPE has been modified to identify and read the input WSSA file from MCNP; it is also able to read the HISTX file produced with HMCNP.

The release of the upgrade package for current LAHET Code System users is intended as a intermediate step before the release of a new LCS installation package including LAHET2.8. HMCNP will cease to exist as a variant of MCNP. Standard MCNP may be fully integrated into the LCS through the use of the standard SSR/SSW options. The upgrade HTAPE is that to be released with LAHET2.8.

Installing the Upgrade Package

The upgrade package is distributed as a UNIX TAR file **upgrade1.tar**. The directory structure is the same as the original LCS installation package. New versions of the files **libhtid** and **libhtix** are included, as are a full set of manual updates (*****.notes**) including two not previously distributed. A set of sample problems are included to illustrate the use of the TRANSM code with MCNP Version 4B. The full contents of the TAR file are as follows:

```
rwxrwxrwx  lcs/aix/
rwx------  lcs/aix/mak_transm
rw-------  lcs/aix/transmi
rwxrwxrwx  lcs/cvx/
rw-------  lcs/cvx/transmi
rwx------  lcs/cvx/mak_transm
rwxrwxrwx  lcs/dec/
rw-------  lcs/dec/transmi
rwx------  lcs/dec/mak_transm
rwxrwxrwx  lcs/fps/
rwx------  lcs/fps/mak_transm
rw-------  lcs/fps/transmi
rwxrwxrwx  lcs/general/
```

```
rw------- 7 lcs/general/libhtid
rw-------  lcs/general/libhtix
rw-------  lcs/general/transmid
rw-------  lcs/general/cinder.notes
rw-------  lcs/general/cylseg.notes
rw-------  lcs/general/damage.notes
rw-------  lcs/general/isosphere.notes
rw-------  lcs/general/multscat.notes
rw-------  lcs/general/xsex.notes
rw-------  lcs/general/bug_fix2.70
rwxrwxrwx  lcs/hpux/
rwx------  lcs/hpux/mak_transm
rw-------  lcs/hpux/transmi
rwxrwxrwx  lcs/sgi/
rwx------  lcs/sgi/mak_transm
rw-------  lcs/sgi/transmi
rwxrwxrwx  lcs/short4b/
rwxrwxrwx  lcs/short4b/1/
rw-------  lcs/short4b/1/inp
rwx------  lcs/short4b/1/short1_script
rw-------  lcs/short4b/1/inh
rw-------  lcs/short4b/1/int
rw-------  lcs/short4b/1/inph
rw-------  lcs/short4b/1/output.ucos
rw-rw-rw-  lcs/short4b/1/output.hpux
rwxrwxrwx  lcs/short4b/1add/
rw-------  lcs/short4b/1add/inp
rwx------  lcs/short4b/1add/short1add_script
rw-------  lcs/short4b/1add/inh
rw-------  lcs/short4b/1add/int
rw-------  lcs/short4b/1add/inph
rw-------  lcs/short4b/1add/output.ucos
rw-rw-rw-  lcs/short4b/1add/output.hpux
rwxrwxrwx  lcs/short4b/4/
rw-------  lcs/short4b/4/inp
rwx------  lcs/short4b/4/short4_script
rw-------  lcs/short4b/4/inh
rw-------  lcs/short4b/4/inph
rw-------  lcs/short4b/4/int
rw-------  lcs/short4b/4/output.ucos
```

```
rw-rw-rw-  lcs/short4b/4/output.hpux
rwxrwxrwx  lcs/short4b/6/
rw-------  lcs/short4b/6/inp
rwx------  lcs/short4b/6/short6_script
rw-------  lcs/short4b/6/inh
rw-------  lcs/short4b/6/inph
rw-------  lcs/short4b/6/int
rw-------  lcs/short4b/6/output.ucos
rw-rw-rw-  lcs/short4b/6/output.hpux
rwxrwxrwx  lcs/sun/
rwx------  lcs/sun/mak_transm
rw-------  lcs/sun/transmi
rwxrwxrwx  lcs/ucos/
rwx------  lcs/ucos/mak_transm
rw-------  lcs/ucos/transmi
rwxrwxrwx  lcs/vms/
rw-------  lcs/vms/transmi.dat
rwx------  lcs/vms/mak_transm.com
```

To install the new features, follow these steps.

- Execute the **mak_libht** script using the new copy of **libhtid**.
- Execute the **mak_transm** script to create TRANSM (requires the **liblht.a** file created in the previous step).
- Test the new code using the sample problems in directory **short4b**.
- One may also recreate LAHET2.70 after inserting the contents of the file **bugfix2.70** into **laheti**. This will fix two "bugs" that may affect LAHET execution.

The TRANSM Code

The TRANSM code plays much the same role as the MRGNTP [2] with one exception. The output file is in the format of an **MCNP "surface source read (SSR)" RSSA file**, which may be input to any recent version of MCNP. As with MRGNTP, file specification information need be given only if the user is defining file names by file replacement. The input file names are the same as for MRGNTP; only the default output file name (RSSA) differs. The default file names are as follows:

NTP1	the first input file
NTP1A	the first continuation of the first file
NTP1B	the second continuation of the first file
NTP1C	the third continuation of the first file
NTP2	the second input file
NTP2A	the first continuation of the second file
NTP2B	the second continuation of the second file
NTP2C	the third continuation of the second file
RSSA	the output file.

The NEUTP files produced by LAHET do not use system familying routines in generating continuations. Consequently, the names may be changed at will to coincide with the MRGNTP/TRANSM default file names (although only three continuations are allowed). The GAMTP and GAMTP1 files produced by PHT have continuation files created by the system familying methods. For these, only the initial file is referenced by MRGNTP or TRANSM; the continuations are automatically accessed by the code. However, the continuation files in this case must be in the user's local file space with the same name as that with which they were written.

Note: in problems with multiple usage of MRGNTP, as in sample problem 1ADD, only the *last* execution of MRGNTP is replaced with TRANSM. For those cases for which an RSSA file is desired for input into MCNP, but no merge is to be done, the following syntax will suffice:

TRANSM NTP1=filename NTP2=0 RSSA=filename

A typical example which combines neutron records from LAHET (NEUTP) with gamma records from PHT (GAMTP) would use the execution line

TRANSM NTP1=NEUTP NTP2=GAMTP RSSA=filename

Using MCNP Version 4B with LAHET

MCNP Version 4B [2] contains both

- all the coding and features of HMCNP4A;
- the new MCNP option to produce a "surface source write" WSSA file to be edited by HTAPE using the SSW output option.

No new coding was required in MCNP to use the volume particle source RSSA file produced by TRANSM. With the new modification to the SSW option, MCNP contains all the features originally developed for HMCNP. To use old HMCNP input files with

MCNP, remove the FILES and FU input records and use the SSR and SSW options as described below,

To use the new source input option, generate an RSSA-type file using TRANSM as described above. Use the SSR source option. **Do not** define units 77, 78, etc. on the FILES input record.

For the new output file option, use the SSW output option in the INP file with the following form:

SSW S_1 S_2 ... S_n SYM=2

Do not define unit 70 on the FILES input record. The option "SYM=2" is new; it causes a surface crossing record to be written for a particle crossing in **either** direction across specified surface S_i. One way to input the WSSA file produced into HTAPE is to use the following syntax:

HTAPE HISTX=WSSA

HTAPE will determine the format of the input file from the header record.

References

1. Richard E. Prael and Henry Lichtenstein, "User Guide to LCS: The LAHET Code System", Los Alamos National Laboratory report LA-UR-89-3014 (September 1989).

2. Judith F. Briesmeister, editor, "MCNP^{TM} - A General Monte Carlo N-Particle Transport Code", Los Alamos National Laboratory report LA-12625-M (March 1997).